A TIME-DEPENDENT SLICE BALANCE METHOD FOR HIGH-FIDELITY RADIATION TRANSPORT COMPUTATIONS

A Thesis
Presented to
The Academic Faculty

by

Steven P. Hamilton

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in
Nuclear Engineering

School of Mechanical Engineering
Georgia Institute of Technology
May 2007

A TIME-DEPENDENT SLICE BALANCE METHOD FOR HIGH-FIDELITY RADIATION TRANSPORT COMPUTATIONS

Approved by:

Dr. Cassiano de Oliveira, Committee Chair
Nuclear and Radiological Engineering
Georgia Institute of Technology

Dr. Weston M. Stacey
Nuclear and Radiological Engineering
Georgia Institute of Technology

Dr. Kevin T. Clarno
Nuclear Science and Technology Division
Oak Ridge National Laboratory

Date Approved: 9 April 2007

ACKNOWLEDGEMENTS

I would like to truly thank all of my committee members for their encouragement and guidance in the preparation of this thesis. I especially want to express my gratitude to Dr. Kevin Clarno for his insight and assistance. My advisor, Dr. Cassiano de Oliveira, has been immensely helpful with his continuous encouragement and unique perspectives. Perhaps most of all, I want to thank Beth for her neverending love and support.

This research was performed under ORNL solicitation 6400006454, Subcontract 4000047518.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

A general finite difference discretization of the time-dependent radiation transport equation is developed around the framework of an existing steady-state three dimensional radiation transport solver based on the slice-balance approach. Three related algorithms are outlined within the general finite difference scheme: an explicit, an implicit, and a semi-implicit approach. The three algorithms are analyzed with respect to the discretizations of each element of the phase space in the transport solver. The explicit method, despite its small computational cost per time step, is found to be unsuitable for many purposes due to its inability to accurately handle rapidly varying solutions. The semi-implicit method is shown to produce results nearly as reliable as the fully implicit solver, while requiring significantly less computational effort.

CHAPTER I

INTRODUCTION

Solutions to the radiation transport equation are of great interest in both the design and operation of reactors. There currently exists a large number of computational tools that have been developed for the analysis of the current fleet of nuclear reactors. These methods tend to make significant approximations in the solution process and account for this through the use of empirically determined correction factors. This approach has led to methods which are extraordinarily fast and accurate within the narrow range of operational experience - predominantly heterogeneous light water reactors. Current trends are leading towards increased heterogeneity in reactor designs, as well as presenting the possibility of new advanced gas-cooled reactors which will display vastly different behavior than the current fleet of reactors [13]. Since experimental data is extremely scarce and operational experience nearly nonexistent, the existing computational approach to reactor analysis will not be adequate for the task at hand. This means that there will be a need for new computational tools that make fewer assumptions and rely less on empirical corrections to perform well.

There are two general classes of methods for solving radiation transport problems: stochastic and deterministic. Stochastic, or Monte Carlo, methods track individual particles from birth until death through the use of probability distributions. Deterministic methods, on the other hand, attempt to formulate a direct mathematical solution to the transport equation. Both Monte Carlo and deterministic methods have their advantages and disadvantages. Monte Carlo methods are able to operate on arbitrary geometries and also may treat angular dependencies without approximation.

Because of this, Monte Carlo methods are generally deemed to be the most accurate radiation transport solvers. However, since the Monte Carlo approach contains an inherent statistical uncertainty, a very large number of particle histories must be simulated in order to produce a reliable solution, resulting in a large computational requirement [7]. Monte Carlo methods tend to excel at producing global, integral quantities such as multiplication factors and reaction rates. If detailed knowledge of the flux distribution of a problem is required, it becomes much more difficult to achieve a converged solution. For problems that contain multiple regions with fissile materials, statistical sampling may average out some very important localized physics, resulting in incorrect solutions despite apparent convergence. One further drawback of the Monte Carlo approach is the difficulty in coupling radiation transport solutions to other types of solvers, such as computational fluid dynamics. Deterministic methods offer several advantages over Monte Carlo methods. A deterministic solution is free of statistical uncertainty and variation, removing one potential pitfall from the solution process, although discretization error is introduced. Detailed knowledge of the neutron flux is automatically produced via the standard solution procedure and coupling to other physics solvers is a much more manageable task.

There is currently a strong movement to develop a computational tool capable of modeling in detail the behavior of a nuclear reactor complete with thermal hydraulic feedback. Such a tool will have a tremendous computational cost associated with it, and it will therefore be necessary for it to be capable of operation in a high performance computing environment. Since there are few radiation transport solvers that are currently capable of producing a detailed flux solution in a parallel computing environment, a high-fidelity transport solver known as NEWTRNX is being developed at Oak Ridge National Laboratory [2]. Prior to this study, the NEWTRNX solver has demonstrated the capability of solving steady-state source driven or eigenvalue problems.

Knowledge of the behavior of a nuclear reactor during a transient situation is vital from both a safety and operational standpoint, yet there are few transport solvers that can perform this task adequately. Many of the parameters in a reactor core design are limited by behavior occurring during either anticipated or unanticipated transients. Without the ability to accurately model these scenarios, large conservative uncertainties must be applied to design analyses. A more accurate depiction of reactor transients would lead to a greater understanding of reactor behavior, which would allow for a decrease in design uncertainties and ultimately the possibility for greater operating efficiency and increased safety margins. This thesis seeks to lay the foundation for a time-dependent radiation transport solver that would serve just such a purpose.

CHAPTER II

THEORY

Of interest are solutions to the Boltzmann radiation transport equation, shown here in time-dependent form with the neutron precursor equation:

$$\begin{aligned} \frac{1}{v}\frac{\partial\psi}{\partial t} + (\Omega\cdot\nabla+\sigma)\psi &= \int dE \int d\Omega\ \sigma_s\psi + (1-\beta)\chi^p \int dE\ \nu\sigma_f\phi \\ &+ S_{ext} + \chi^d \sum_{i=1}^{6} \lambda_i C_i \qquad (2.1) \\ \frac{\partial C_l}{\partial t} &= \beta_l \int dE \nu\sigma_f\phi - \lambda_l C_l \end{aligned}$$

where $\psi(\vec{r},\hat{\Omega},E,t)$ is the angular flux, $v(E)$ is the neutron speed, $\hat{\Omega}$ is the direction of neutron travel, $\sigma(\vec{r},E)$ is the total (macroscopic) interaction cross section, $\sigma_s(\hat{\Omega}\rightarrow\hat{\Omega}',E'\rightarrow E)$ is the scattering cross section, β is the total delayed neutron fraction, $\chi^p(E)$ is the prompt fission spectrum, $\nu\sigma_f(\vec{r},E)$ is the number of neutrons release per fission multiplied by the fission cross section, $\phi(\vec{r},E,t)=\int d\hat{\Omega}\psi(\vec{r},\hat{\Omega},E,t)$ is the scalar flux, $S_{ext}(\vec{r},\hat{\Omega},E,t)$ is the external source, $\chi^d(E)$ is the delayed neutron fission spectrum, and $C_l(\vec{r},t)$, β_l, and λ_l are the delayed neutron precursor concentration, the delayed neutron fraction, and the decay constant, respectively, for delayed group l.

The quantity of interest, the angular flux, depends on a total of seven independent variables (three in space, two in angle, energy, and time). This substantial variable dependence, along with the integrals over energy and angle on the right hand side of the equation, make the task of solving the transport equation very formidable. In order to transform this problem into one that may be readily solved numerically, the entire phase space must be discretized. It is common practice in transport calculations to consider a steady state situation in which the time derivative is set to zero and the

neutron precursors therefore have no impact [14], in which case equation 2.1 becomes:

$$(\Omega \cdot \nabla + \sigma)\psi = \int dE \int d\Omega \; \sigma_s \psi + \chi^p \int dE \; \nu\sigma_f \phi + S_{ext} \tag{2.2}$$

Various discretizations have been used in transport solvers, especially with respect to the space and angular discretizations. The discretizations used in the NEWTRNX transport solver are slice-balance in space-angle, multigroup in energy, and finite difference in time. The essential aspects of these discretizations will now be discussed.

2.1 General Considerations

The most common strategy for handling the integrals in equation 2.2 is a process known as source iteration. The approach is to assume a value for the angular flux, and use that value to calculate the right hand side of equation 2.2, which is generally written as a single source term:

$$Q = \int dE \int d\Omega \; \sigma_s \psi + \chi^p \int dE \; \nu\sigma_f \phi + S_{ext} \tag{2.3}$$

Equation 2.2 then becomes:

$$(\Omega \cdot \nabla + \sigma)\psi = Q \tag{2.4}$$

Now with a constant right hand side, the equation is solved to produce an updated value for the angular flux. This new flux is used to update the source term and the process is repeated until two successive iterates converge to within some specified tolerance [11] .

2.2 Space-Angle Discretizations

Due to the Slice Balance Approach (SBA) used in NEWTRNX, the discretizations of the space and angle variables are intrinsically linked. SBA is a solution strategy that allows for calculations to be performed on a general 3D geometry within the bounds of a discrete ordinates methodology. In general, any volume is first approximated by an arbitrary polyhedral element that conserves volume. The vital characteristic of

the SBA is the dual spatial discretization: a coarse, angle-independent mesh upon which the right hand side of equation 2.2 is evaluated, and an angle-dependent decomposition within each coarse cell on which the left hand side of equation 2.2 is evaluated [8]. The selection of angles in SBA is the same as in any discrete ordinates method: a quadrature set is used to evaluate the angular integral in equation 2.3. By requiring the transport equation to only hold for some finite number of angles, the angular integral in 2.3 reduces to a weighted sum over the selected angles. For the isotropic scattering component this becomes:

$$\int d\hat{\Omega}\sigma_s(E' \rightarrow E)\psi(E') = \sum_{m=1}^{M} w_m \sigma_s(E' \rightarrow E)\psi_m(E') \tag{2.5}$$

where w_m are the weights for the integration quadrature. Similar expressions involving spherical harmonic terms can be written for the anisotropic expansion of the scattering cross section [14]. A discussion of the merits of various quadrature sets is out of the scope of this study.

The SBA strategy is based on the characteristics form of the transport equation:

$$\frac{d\psi(s)}{ds} + \sigma(s)\psi(s) = Q(s) \tag{2.6}$$

where the equation is defined along a particular (characteristic) direction, and s is the distance from some reference point along that direction. A simplification can be made if it is assumed that the source term, Q, and the total cross section, σ, are constant within a region of interest. This is known as the Step Characteristics Approximation [7]. In this case equation 2.6 can be solved analytically in terms of the solution at s=0, ψ_0:

$$\psi(s) = \psi_0 e^{-\sigma s} + \frac{Q}{\sigma}(1 - e^{-\sigma s}) \tag{2.7}$$

In the SBA, this solution is only extended across a single computational cell. This is in contrast to solvers that employ the Method of Characteristics, where the characteristic is extended across the entire problem domain [7].

The angle-dependent decomposition of a single coarse mesh cell takes place as follows. The cell is divided into some number of subregions called slices. Each of these slices contains a single incident and exiting face and a number of faces which are parallel to the characteristic direction. This process is difficult to illustrate in 3D, but figure 1 shows the result in an analogous 2D situation.

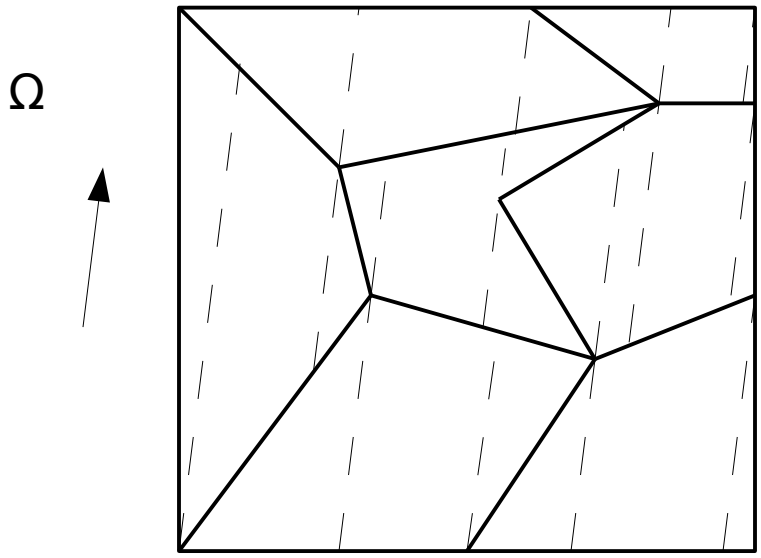

Figure 1: Slicing of a 2D Mesh

Each main region is divided into a number of slices based on the particular direction of interest, and the slicing of each cell is independent of adjacent cells. For the angle shown, the left and bottom exterior boundaries would require specified boundary conditions and the solution would proceed from bottom left to upper right until reaching the top and right boundaries. In many unstructured mesh solvers, concave cells cause difficulties by producing cyclic dependencies. In the slice balance method, however, the sweep across the domain is not based upon cells but rather slices, which cannot be concave. Cyclic dependencies, therefore, are essentially eliminated [8] . In

a three dimensional situation, each slice takes the form of a tube in which all the faces are parallel to the direction of interest except a single incident and exiting faces.

By defining an average solution on the incident face, the solution of equation 2.7 may be used to form an analytic solution for the flux averaged over the volume and also over the exiting face for a single slice [8]. Since each coarse mesh cell contains multiple slices, the solutions on the slices must be combined to form a single solution on the coarse mesh. This is accomplished by weighting the slice solutions with their respective volumes or areas. The volume averaged flux in the coarse cell is the sum of the volume fractions multiplied by the slice averaged flux:

$$\psi_{cell} = \frac{1}{V_{cell}} \sum_{slice \in cell} V_{slice} \psi_{slice} \tag{2.8}$$

Likewise, the flux on a given face of the coarse cell is the sum of the area fractions of the slice faces composing the face multiplied by the average flux on the corresponding slice:

$$\psi_J = \frac{1}{A_J} \sum_{face \in J} A_{face} \psi_{face} \tag{2.9}$$

where J indicates a face of the coarse cell. After the solution has been constructed across the entire domain for each angle of the quadrature set, the source term of equation 2.3 is evaluated based on the quadrature formula of equation 2.5 or an equivalent formulation taking into account anisotropic scattering.

2.3 Energy Discretization

The energy discretization that has been implemented in NEWTRNX is the standard multigroup treatment commonly used in deterministic radiation transport. According to the multigroup formulation, the continuum of energy values are discretized into a number of energy groups and all material properties are assumed to be constant within any given energy group. In order for this approximation to perform well, all quantities of interest must be appropriately averaged over a given energy group. For the angular

flux, this is achieved in a straightforward manner by integrating over the energy group: $\psi_g = \int_{E_{g-1}}^{E_g} dE\ \psi(E)$, where the subscript g indicates a particular energy group. For the cross-sections, however, such a naïve treatment would not produce accurate results. Instead, the cross-sections must take into account the physics that occurs within the energy group. This is normally achieved by formulating the cross-sections to preserve the reaction rate within the energy group: $\sigma_{xg}\psi_g = \int_{E_{g-1}}^{E_g} dE\ \sigma_x(E)\psi(E)$, where x indicates any reaction. This leads directly to the definition of the multigroup cross-section:

$$\sigma_{xg} = \frac{\int_{E_{g-1}}^{E_g} dE\ \sigma_x(E)\psi(E)}{\int_{E_{g-1}}^{E_g} dE\ \psi(E)} \tag{2.10}$$

The problem with this definition is that it requires knowledge of the detailed solution for the angular flux, $\psi(E)$, but this is the desired result of the entire computation and is therefore unknown. So instead, an approximate value of the angular flux is used as a weighting function in equation 2.10 [11]. A description of the weighting functions used in generating multigroup cross sections is out of the scope of this discussion and may be found elsewhere.

2.4 Time Discretization

The time discretization that has been implemented in NEWTRNX is a finite difference scheme - the time derivatives in 2.1 are represented by a difference between consecutive time points:

$$\begin{aligned} \frac{\partial\psi}{\partial t} &= \frac{\psi^{i+1} - \psi^i}{\Delta t} \\ \frac{\partial C_l}{\partial t} &= \frac{C_l^{i+1} - C_l^i}{\Delta t} \end{aligned} \tag{2.11}$$

Inserting this into equations 2.1 and rearranging leads to:

$$\begin{aligned} \left(\hat{\Omega}\cdot\nabla + \sigma + \frac{1}{v\Delta t}\right)\psi^{i+1} &= Q^\alpha + \sum_{l=1}^{6}\lambda_l C_l^\alpha + \frac{\psi^i}{v\Delta t} \\ C_l^{i+1} &= C_l^i + \Delta t(\beta_l \nu\sigma_f\phi^\alpha - \lambda_l C_l^\alpha) \end{aligned} \tag{2.12}$$

Defining a modified time-dependent total cross section and source term:

$$\begin{aligned} \tilde{\sigma} &= \sigma + \frac{1}{v\Delta t} \\ \tilde{Q} &= Q + \sum_{l=1}^{6} \lambda_l C_l + \frac{\psi^i}{v\Delta t} \end{aligned} \tag{2.13}$$

the first of equations 2.12 becomes:

$$(\Omega \cdot \nabla + \tilde{\sigma})\psi^{i+1} = \tilde{Q}^{\alpha} \tag{2.14}$$

This is essentially identical to the steady state problem of equation 2.2 [9]. The superscript α indicates that the quantity is evaluated at a yet-undetermined point in time. By selecting different values of α, various algorithms may be developed, each with unique properties. The simplest choice of is to evaluate all of the α terms in the first of equations 2.12 at the previous time step, $\alpha = i$. Since these values are already known, the calculation at each time step is equivalent to solving a steady-state fixed source problem in a purely absorbing medium. This calculation may be performed very quickly and the computational requirement for each time step is small. In addition, since equation 2.14 is solved successively for each energy group, it is actually possible to use an updated flux value to calculate the scattering source contribution from all higher energy groups, a modification analogous to performing a Gauss-Seidel rather than a Jacobi iteration in energy. This is termed a semi-implicit approach by Lewis and Miller [11] and an explicit approach by Hill and Reed [9]. In this study, it will be referred to as an explicit approach. This is the approach used in the 1-D transport solver TIMEX, which has been shown to require very small time steps for accurate solutions [3].

The next logical choice for α is to evaluate all of the terms at the next time step, $\alpha = i + 1$. Since the values at the next time step are not known, the source iteration process described earlier must be performed at each time step. This is equivalent to solving a steady-state fixed source problem, but it is now necessary to iterate on

the scattering and fission sources. In highly scattering media, this process is not trivial and the overall computational cost for this solution approach can therefore be quite large. This method will be referred to as an implicit algorithm. The implicit algorithm is the solution method implemented in the EVENT radiation transport solver [5].

One additional possibility is to begin the implicit approach, but prematurely stop the iterations after either a fixed number of iterations or a relaxed convergence criterion is met. For this case, α does not directly correspond to either i or $i+1$, but rather falls somewhere between the two. In this semi-implicit approach, information is passed more rapidly than with the explicit algorithm, yet the computational cost is not as high as for the implicit algorithm. Since a great deal of freedom is available in the selection of the iteration or convergence criteria, this approach actually represents a continuum of possible algorithms ranging from fully explicit to fully implicit.

The selection of α for the second of equations 2.12 is a much simpler task. The lack of any spatial derivatives means that the equation may be inverted trivially, allowing the selection of $\alpha = i+1$ at no additional computational cost. Thus, regardless of the solution scheme for the first of equations 2.12, the second equation may be treated implicitly. Additionally, the updated flux from the solution of the first equation may be used to evaluate the production rate in the precursors equation. An investigation into the merits of the algorithms described in this section will be the primary purpose of this study.

2.5 Analytical Solutions

If a neutron population is independent of all variables other than time, an analytic solution may be produced for the time variation of the flux. Working in terms of the neutron density, $n(t) = \frac{\psi(t)}{v}$, and defining the prompt neutron lifetime $\Lambda = (v\nu\sigma_f)^{-1}$

and the reactivity $\rho = \frac{\delta k}{k}$, equations 2.1 can be reduced to the point kinetics equations:

$$\frac{dn(t)}{dt} = \frac{\rho - \beta}{\Lambda} n(t) + \sum_{l=1}^{6} \lambda_l C_l(t) \tag{2.15}$$

$$\frac{dC_l(t)}{dt} = \frac{\beta_l}{\Lambda} n(t) - \lambda_l C_l(t)$$

By Laplace transforming these equations, a seventh order equation known as the inhour equation is produced:

$$Y(s) = \rho - s\left(\Lambda + \sum_{l=1}^{6} \frac{\beta_l}{s + \lambda_l}\right) \tag{2.16}$$

There are seven roots to this equation which lead to a solution of the form:

$$n(t) = \sum_{j=0}^{6} A_j e^{s_j t} \tag{2.17}$$

The coefficients, Aj, are given by [1]:

$$A_j = \frac{\left(\Lambda + \sum_{l=1}^{6} \frac{\beta_l}{s + \lambda_l}\right)}{\left(\lambda + \sum_{l=1}^{6} \frac{\beta_l \lambda_l}{(s + \lambda_l)^2}\right)} \tag{2.18}$$

If it is further assumed that the behavior of the delayed neutrons can be represented by a single effective delayed neutron group, then the equivalent of the inhour equation has only two roots and an approximation to the resulting solution appears as:

$$n(t) = n_0 \left[\frac{\rho}{\rho - \beta} exp\left(\frac{\rho - \beta}{\Lambda} t\right) - \frac{\beta}{\rho - \beta} exp\left(-\frac{\lambda \rho}{\rho - \beta} t\right)\right] \tag{2.19}$$

The first exponential term is generally very rapidly decaying. This leads to a portion of the solution that changes extremely quickly after a perturbation to a system, a behavior that is known as the prompt jump. The short time period associated with the prompt jump generally leads to a need to achieve very high temporal resolution shortly after a system is altered. The second exponential term presents a much more gradual change in the solution, leading to the asymptotic solution that prevails at time periods beyond the prompt jump. For a system located in this regime, resolution requirements are far less stringent.

CHAPTER III

ANALYSIS/RESULTS

Though the time-marching algorithms presented in the previous section are very similar in nature, they tend to exhibit unique behavior under certain circumstances. The impact of each of the various phase-space discretizations on the algorithms will be analyzed. The methods will be compared against each other to identify pertinent details of the different algorithms. Where possible, solutions will be compared against analytical solutions to verify that the expected results are being obtained.

3.1 Time Behavior

To illustrate the fundamental features of the time-dependent algorithms, the first problem will seek to recreate the situation of a one speed solution in an infinite homogeneous medium. In this case, the most pertinent features of the various algorithms will be clearly exhibited and a direct comparison to analytical solutions can be performed. The solutions of the point kinetics equations presented in chapter 2 will serve as the standard for comparison in these tests.

Within NEWTRNX, an infinite homogeneous problem is simulated via a cubic region with reflecting boundary conditions on all sides. First, the case of a negative reactivity insertion corresponding to an immediate change from a critical system to a k_∞ value of 0.95 is considered. For this case, a single delayed neutron group is considered with $\beta = 0.0075$ and $\lambda = 0.08s^{-1}$, parameters typical of a thermal reactor composition. Figure 2 displays a comparison of the solutions for each algorithm as compared to the analytical solution of equation 2.19. For a sufficiently small time step, all three methods will converge to the same result, which is identical to the analytical solution. When a larger time step is used, however, discrepancies appear

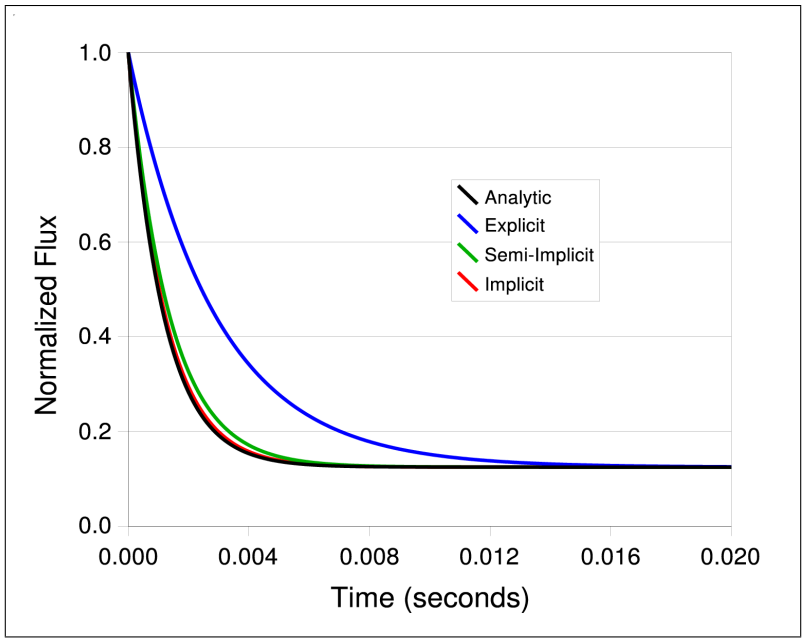

Figure 2: Infinite Media Solutions for a Negative Transient with One Delayed Neutron Group

between the solutions. Most notably, the explicit solution is unable to vary as rapidly as the actual solution due to the fact that the neutron balance is not fully satisfied in this case [9]. The fully implicit method does not suffer from this drawback and is therefore able to reproduce the actual solution quite well for a much larger time step. The semi-implicit approach, as would be expected, lies somewhere in between the other two approaches. It is not able to follow the correct solution exactly, but it does extremely well for even a small number of iterations per time step. In this and the remaining analyses within this study, the semi-implicit approach consists of performing four outer (source update) iterations at each time step. At times greater than the time duration of the prompt jump, all three solutions appear to converge to the actual asymptotic solution. If the results are examined in great detail, however, it is observed that the explicit and semi-implicit methods actually approach a slightly higher asymptotic value than the implicit method. This is because the two methods actually accumulate a higher precursor concentration over the course of the prompt

jump due to the higher intermediate fluxes. Table 1 shows the maximum and final values of the error during this transient for each of the algorithms and two different time steps and a rough estimate of the computational effort via the number of outer iterations performed.

Table 1: Calculated Error for a Negative Transient

Δt	Algorithm	Maximum Error	Error at $t = 0.2$	Iterations Per Time Step
0.001	Explicit	479%	3.1%	1.0
	Semi-Implicit	240%	0.052%	4.0
	Implicit	36.4%	0.0054%	22.9
0.0001	Explicit	128%	0.021%	1.0
	Semi-Implicit	16.1%	0.0003%	4.0
	Implicit	4.1%	0.0019%	4.6

The explicit method performs one outer iteration per time step by definition and the semi-implicit solution similarly performs four iterations per time step. The implicit solution only requires slightly more iterations than the semi-implicit to converge for the smaller time step, but for the larger time step the implicit approach requires many times more iterations. Overall, even the explicit method is seen to perform well unless a very high level of accuracy or a very detailed account of the prompt jump is required.

Next, the behavior corresponding to an immediate change from a critical configuration to a system with a k_∞ of 1.0015 with six delayed neutron groups is considered. The delayed neutron parameters, typical of ^{235}U, are shown in table 2. Figure 3 illustrates the calculated behavior during a transient corresponding to a positive reactivity insertion with six delayed precursor groups. The standard for comparison in this instance is the inhour solution of equation 2.16, where the roots and coefficients have been calculated numerically to a high level of precision. Similar behavior is again observed for the algorithms during the positive transient. The explicit method is unable to grow as rapidly as is necessary, and therefore produces a solution that

Table 2: Six Group Delayed Neutron Parameters

Group	β_l	$\lambda_l(s^{-1})$
1	2.48×10^{-4}	0.0124
2	1.64×10^{-4}	0.0305
3	1.47×10^{-3}	0.111
4	2.96×10^{-3}	0.301
5	8.63×10^{-4}	1.14
6	3.15×10^{-4}	3.01
Total	7.50×10^{-3}	–

is somewhat less than the actual result. The semi-implicit approach also exhibits this behavior, though to a much lesser extent, and the implicit method has no such difficulties. Since the explicit and semi-implicit methods produce a solution that is lower than the exact value during the prompt jump, they also indicate a slightly lower production of delayed precursors during that time period and therefore converge to a slightly lower asymptotic solution at times beyond the prompt jump. These effects are displayed in table 3. Whereas the discrepancy for the negative transient was conservative, for the positive transient the deviation is non-conservative. The relative computational effort between the three methods is similar to the previous case, though the implicit solution must work slightly harder for the positive transient.

Table 3: Calculated Error for a Positive Transient

Δt	Algorithm	Maximum Error	Error at $t = 0.2$	Iterations Per Time Step
0.001	Explicit	16.3%	7.9%	1.0
	Semi-Implicit	10.4%	0.97%	4.0
	Implicit	0.36%	0.025%	29.3
0.0001	Explicit	6.80%	0.31%	1.0
	Semi-Implicit	1.1%	0.029%	4.0
	Implicit	0.045%	0.0081%	5.6

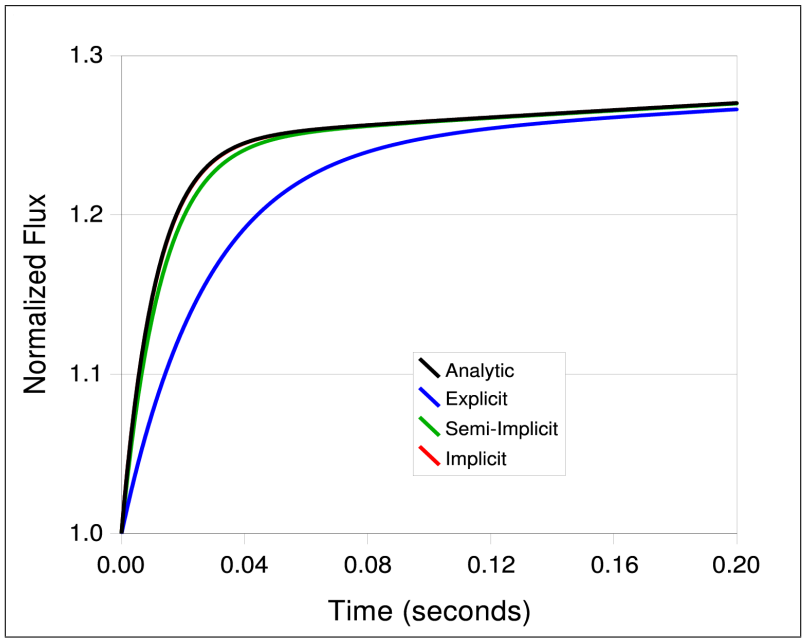

Figure 3: Infinite Media Solutions for a Positive Transient with Six Delayed Neutron Groups

3.2 Energy-Time Behavior

In order to generalized the previous results somewhat, a scenario will now be considered in which the angular flux varies with energy and time, while still remaining independent of space and angle. In order to illustrate the impact of the time discretizations on the energy distribution, we will look at a case where a source of neutrons in the highest energy group is pulsed for a short duration and then removed. The material is a homogeneous mixture of ^{235}U and hydrogen which has an infinite medium multiplication factor of approximately $k_\infty = 0.956$. The cross sections utilize the SCALE [12] 27-group structure with 16 fast and 11 thermal groups and were produced using the TRITON sequence [6]. In this example, the neutron source is pulsed for a period of 2.5 microseconds and then removed. The responses for a typical fast, intermediate, and thermal group (groups 2, 8, and 24, respectively) with each algorithm are shown in figures 4, 5, and 6, respectively, for a time step of 0.1 microseconds. The fast group displays a prompt jump then slowly increases while

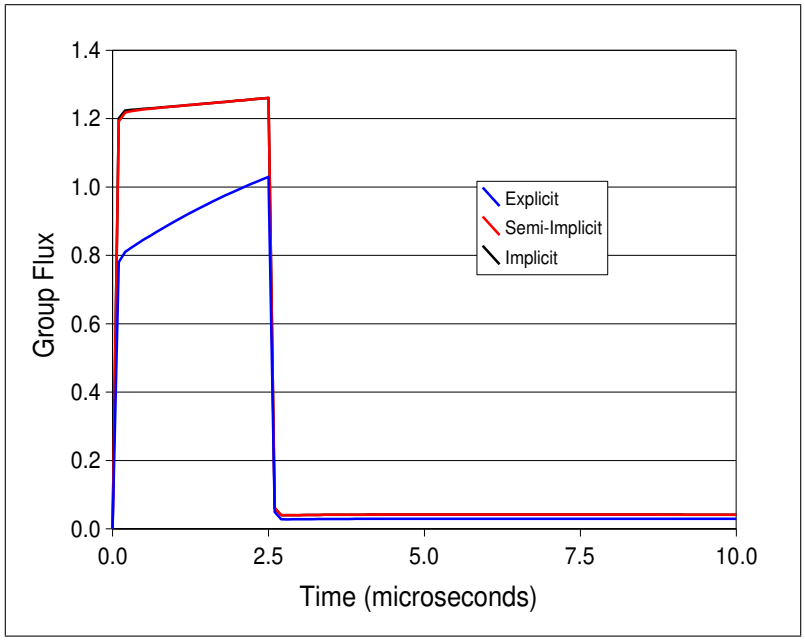

Figure 4: Group 2 Response to a Pulsed Neutron Source with 2.5 microsecond Duration

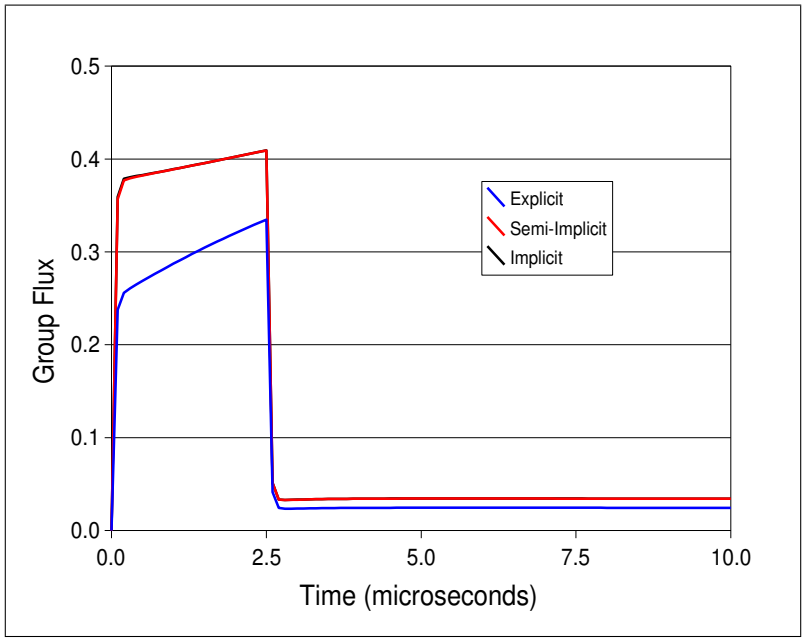

Figure 5: Group 8 Response to a Pulsed Neutron Source with 2.5 microsecond Duration

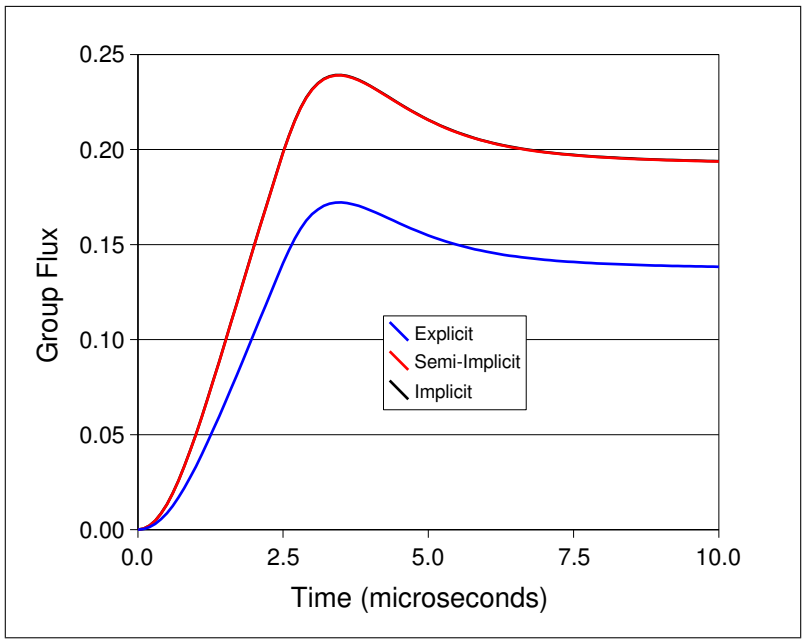

Figure 6: Group 24 Response to a Pulsed Neutron Source with 2.5 microsecond Duration

the source is turned on, then drops back down almost to zero as soon as the source is removed. The intermediate group shows nearly the same behavior, but with a smaller magnitude than the fast group. The thermal group responds slowly to the source, then gradually grows as neutrons are thermalized and does not reach a maximum until after the source has been removed. At long times after the source removal, the flux slowly decays away with a period corresponding to the decay constant of the delayed neutrons.

Comparing the algorithms, the semi-implicit and implicit approaches display essentially identical performance, and are indistinguishable on figures 4, 5, and 6. The explicit approach, however, again is unable to keep up with the rapidly changing flux in the fast group and exhibits significant deviations from the desired result. Whereas in the previous section the explicit solution would eventually catch up to the actual value, in this case the source is removed before this can occur and the inaccuracies are propagated through the duration of the calculation. The errors for the explicit approach in the intermediate and thermal groups are strictly due to the inability to model the fast group correctly.

Figures 7 and 8 show the normalized energy spectrum shortly after the source is initially inserted and shortly after it is removed, respectively. At a time of 0.5 microseconds after the source insertion, the spectrum is dominated by the high energy source. The shortcomings of the explicit approach cause it to produce a slightly faster spectrum than the semi-implicit or implicit methods, though the discrepancy is fairly small. Shortly after the removal of the source, the spectrum closely resembles a typical thermal neutron spectrum with a pronounced Maxwellian distribution in the low energies and a fission spectrum at high energies. The explicit algorithm again produces a slightly faster spectrum than the other methods, owing to its inability to capture the negative transient in the fast groups. In general, all three methods do at least an adequate job of reproducing the shape of the energy spectrum during a

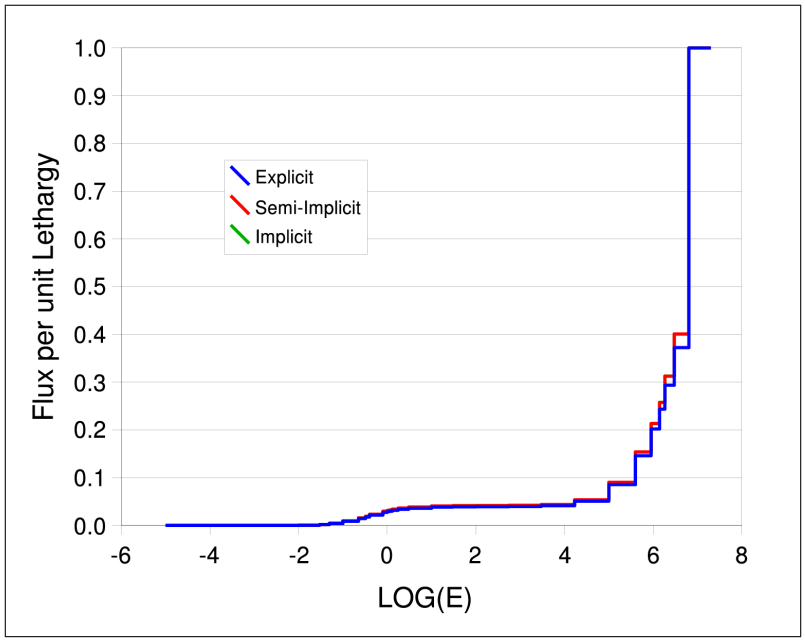

Figure 7: Energy Spectrum $0.5\mu s$ After Insertion of Source

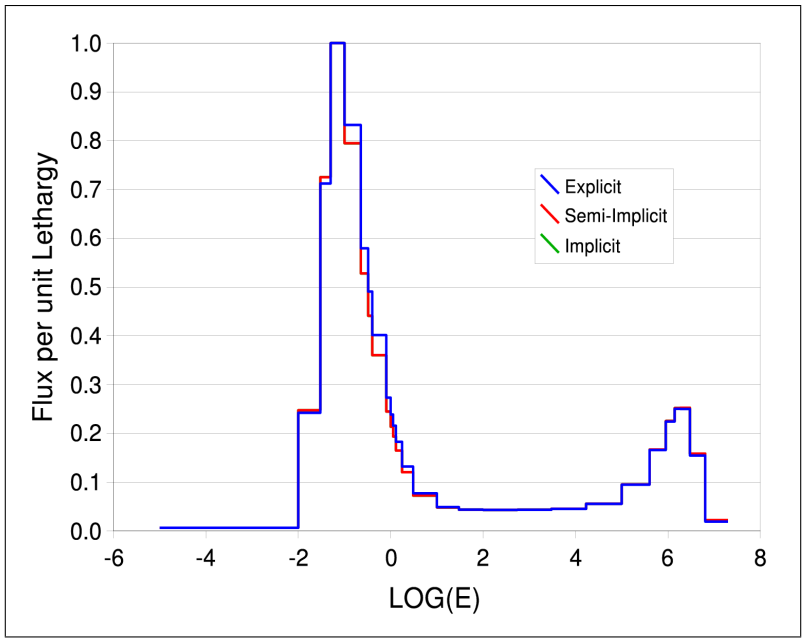

Figure 8: Energy Spectrum $0.5\mu s$ After Removal of Source

transient, it is only the calculation of the magnitudes of the fluxes where the explicit method greatly suffers. In this analysis, it has again been observed that the semi-implicit approach is able to reproduce almost all of the detailed behavior evident in the implicit approach, while requiring significantly less computational effort.

3.3 Space-Time Behavior

A final test of the implementation of the time-dependent algorithms involves the solution of a problem with a significant time-varying spatial distribution. In order to accomplish this, a problem is established that contains a region of fissile material in a slightly supercritical configuration but starting with a zero neutron population. At $t = 0$ a source is started in a region away from the fissile material. This configuration is shown in figure 9. Vacuum boundaries are present on all four sides. The geometry is uniform in the direction normal to the page with reflecting boundaries on the top and bottom, creating an effective two dimensional region. As the source is turned on, the neutrons begin to spread out across the domain. After a few milliseconds, the flux has reached a high enough level in the fissile region to begin a significant chain reaction. The solution then begins to grow in an exponential fashion. In this study, the calculation is terminated after 5 milliseconds, as beyond that point the solution simply continues to grow rapidly. Table 4 displays the error relative to the implicit solution and the relative computational effort for the three solution methods with a time step of $50\mu s$. Since the errors are being measured relative to the implicit solution, the error of that method is identically zero. The maximum error for both

Table 4: Algorithm Performance for a 2-D Problem

Algorithm	Maximum Error	Iterations Per Time Step
Explicit	89%	1.0
Semi-Implicit	16%	4.0
Implicit	0%	11.8

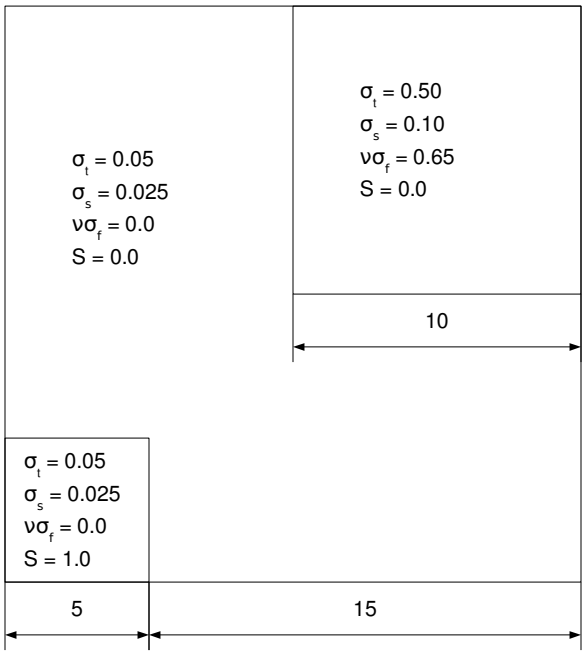

Figure 9: Materials and Geometry for Two Dimensional Problem

the explicit and semi-implicit approaches occurs within the fissile region at the final time step. This is consistent with earlier observations that indicated those methods have difficulty accurately representing rapidly varying solutions. Outside of the fissile region, all three solution methods behave very well, and the associated errors are generally less than 1% for the semi-implicit solution and 5% for the explicit. Figures 10, 11, and 12 show the solution at $t = 2.0\mu s$ for the three solution methods. Very good agreement can be seen between the implicit and semi-implicit methods. The explicit solution does quite well at capturing the behavior in the vicinity of the source region, but due to the rapid changes that are occurring it does not fare as well in the fissile region and the growth of the flux is not yet evident.

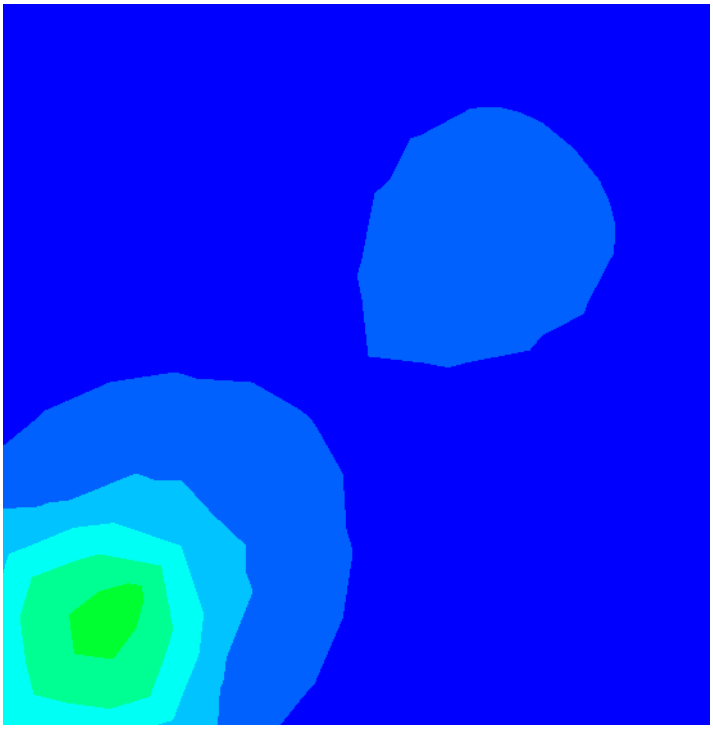

Figure 10: Implicit Solution of Two Dimensional Problem at $t = 2.0\mu s$

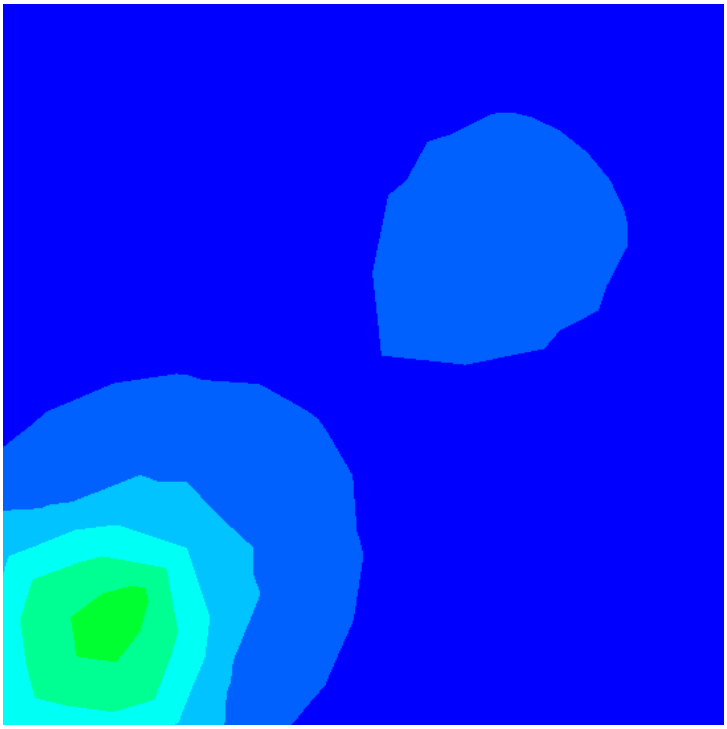

Figure 11: Semi-Implicit Solution of Two Dimensional Problem at $t = 2.0\mu s$

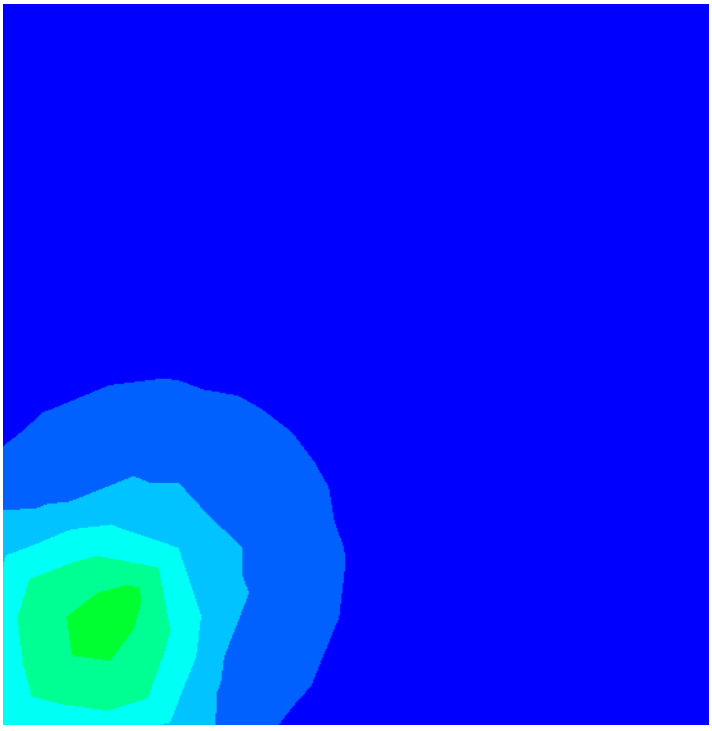

Figure 12: Explicit Solution of Two Dimensional Problem at $t = 2.0\mu s$

Figures 13, 14, and 15 show the flux distributions at a later time, $4.0\mu s$ after the beginning of the problem. The semi-implicit solution again only slightly deviates from the implicit solution and the explicit solution still displays a noticable lag with respect to the other algorithms. Upon consideration of a spatially dependent problem, the crucial aspects of the solution schemes remain unchanged. The semi-implicit solution appears capable of capturing the vast majority of the physics, while the explicit algorithm does not perform this task nearly as well.

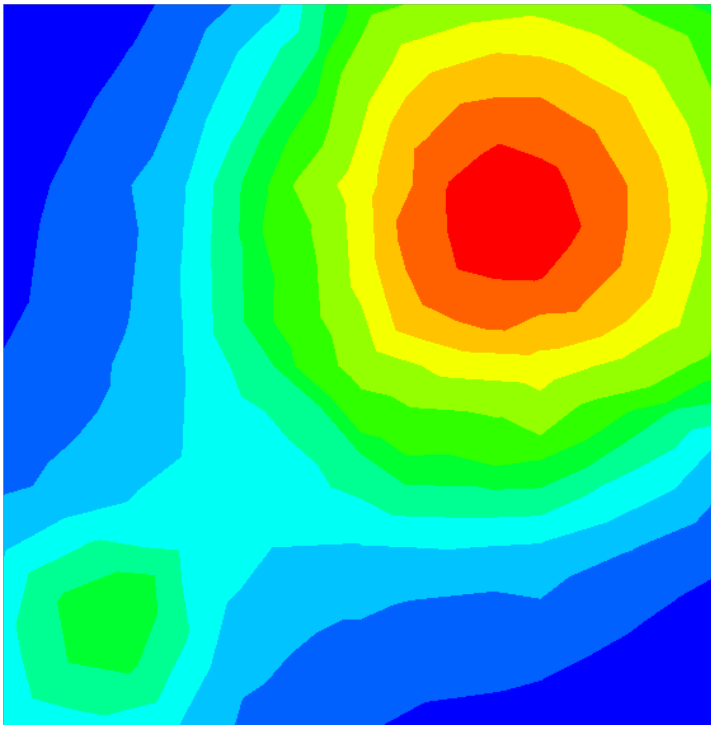

Figure 13: Implicit Solution of Two Dimensional Problem at $t = 4.0\mu s$

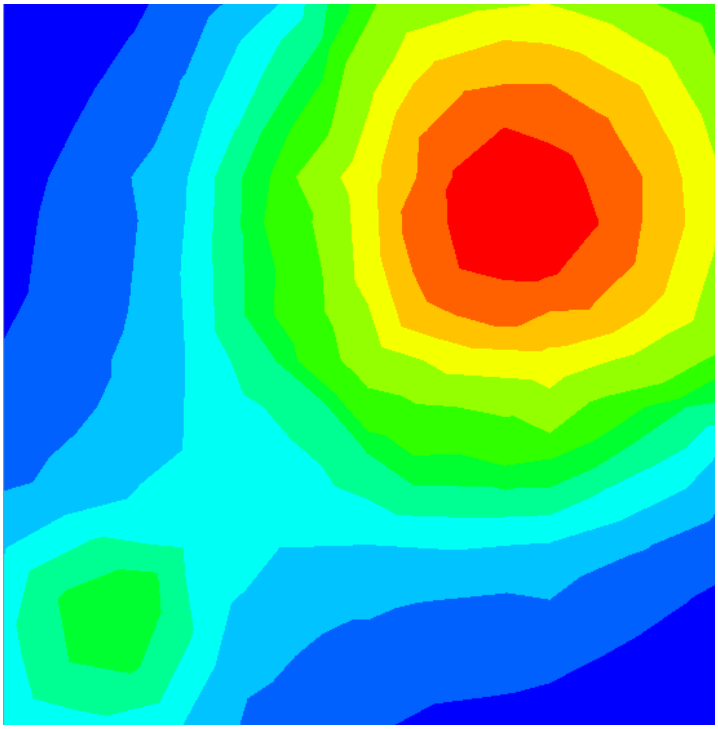

Figure 14: Semi-Implicit Solution of Two Dimensional Problem at $t = 4.0\mu s$

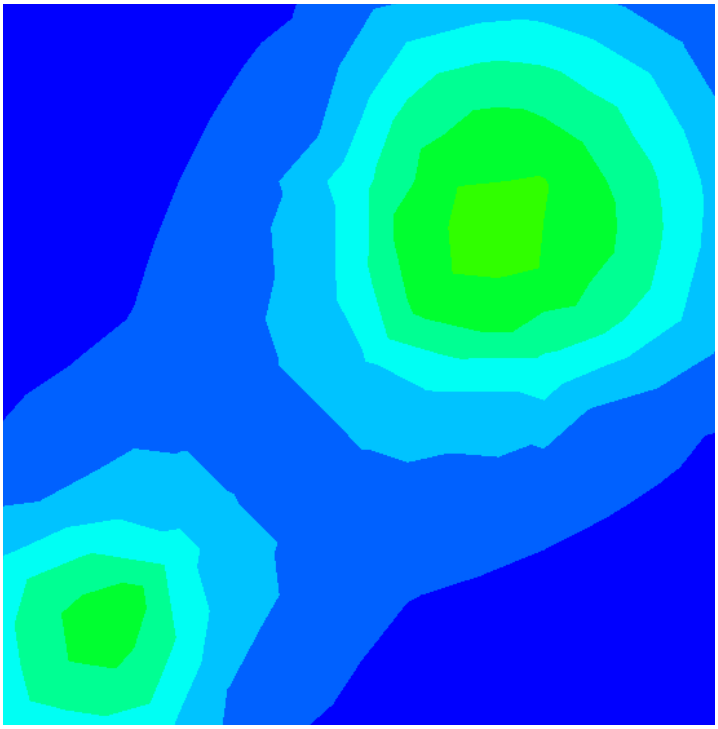

Figure 15: Explicit Solution of Two Dimensional Problem at $t = 4.0\mu s$

CHAPTER IV

CONCLUSION

A general formulation for a time-dependent radiation transport solution scheme has been developed around the foundation of an existing steady-state transport solver. Three different algorithms have been presented, each with its own set of strengths and weaknesses. The explicit approach, while requiring minimal computational effort, is generally unable to accurately represent rapidly changing fluxes. A very small time step is therefore required in order to obtain accurate solutions. The implicit algorithm allows for significantly larger time steps to be used to produce results with the same accuracy. The cost is that a much more computationally intensive problem must be solved for each time step, greatly increasing the expense of a single time step. The semi-implicit algorithm is able to draw upon the strengths of the other two methods, producing an algorithm that has neither an overly stringent restriction on the allowable time step nor an excessive computational cost per time step.

As this study is intended to be only the first step in the production of a tool capable of simulating the behavior of a nuclear reactor, significant work remains in the future for improving the time-dependent capabilities of NEWTRNX. In the near term, it is desired to develop some strategies to improve the efficiency of the solver to minimize the computational time necessary to perform a computation. Potential strategies for this goal include incorporating an adaptive time stepping procedure, implementing some form of an exponential extrapolation such as that formulated by Hill and Reed [9], and a higher order solution approach, such as a Crank-Nicholson strategy. The latter requires an implicit algorithm, but yields a solution that is second order accurate in time. If some of the known drawbacks of that method, such

as non-physical oscillations in the solution [4], can be mitigated then the extra cost per time step might be offset by the increase in step size that could be tolerated. An additional area for work is to extend the current time-dependent capabilities to account for reactor dynamics taking place on a longer time scale. This would include coupling the existing solver with a depletion code such as ORIGEN-S [10] to automate the changing isotopic distribution during reactor operation. This is necessary to take into account the impact of fission products and burnup, which have a very large effect on the behavior of a reactor. Looking even further ahead, the coupling of NEWTRNX with a computational fluid dynamics solver will be necessary to simulate the thermal hydraulic feedback within a reactor.

REFERENCES

[1] Bell, G. and Glasstone, S., *Nuclear Reactor Theory*. Van Nostrand Reinhold Company, 1970.

[2] Clarno, K. T., "GNES-R: Global nuclear energy simulator for reactors. task 1: High-fidelity neutron transport," in *Reactor Physics Division Topical Meeting: PHYSOR*, 2006.

[3] de Oliveira, C. R. E., "A time-dependent finite element method: Progress report for the first six months of the contract," Tech. Rep. NNS/32A/1A92358, Queen Mary College, 1989.

[4] de Oliveira, C. R. E., "A time-dependent finite element method: Progress report to june 1989," Tech. Rep. NNS/32A/1A92358, Queen Mary College, 1989.

[5] de Oliveira, C. R. E. and Goddard, A., "EVENT - a multidimensional finite element spherical harmonics radiation transport code," in *Proceedings of Int. Seminar 3-D Deterministic Radiation Transport Codes*, (Paris, France), Organization for Economic Cooperation and Development, 1996.

[6] DeHart, M. D., *TRITON: A Two-Dimensional Depletion Sequence for Characterization of Spent Nuclear Fuel*. NUREG/CR-200, Rev. 7, Vol I Section T1, ORNL/NUREG/CSD-2/R7, 2004.

[7] DeHart, M. D., "Advancements in generalized-geometry discrete ordinates transport for lattice physics calculations," in *Proc. of PHYSOR*, (Vancouver, British Columbia), 2006.

[8] Grove, R. E., "The slice balance aproach (SBA): A characteristic-based, multiple balance SN approach on unstructured polyhedral meshes," in *Proceedings of the Mathematics and Computation, Supercomputing, Reactor Physics and Nuclear and Biological Applications*, (La Grange Park), 2005.

[9] Hill, T. R. and Reed, W. H., "TIMEX: A time-dependent explicit discrete ordinates program for the solution of multigroup transport equations with delayed neutrons," Tech. Rep. LA-6201-MS, Los Alamos Scientific Laboratory, 1976.

[10] I. C. Gauld, O. W. H. and Westfall, R. M., *ORIGEN-S: SCALE System Module to Calculate Fuel Depletion, Actinide Transmutation, Fission Product Buildup and Decay, and Associated Radiation Source Terms*. NUREG/CR-200, Rev. 7, Vol I Section T1, ORNL/NUREG/CSD-2/V1/R7, 2004.

[11] Lewis, E. E. and W. F. Miller, J., *Computational Methods of Radiation Transport*. La Grange Park: American Nuclear Society, 1993.

[12] SCALE, *A Modular Code System for Performing Standardized Computer Analyses for Licensing Evaluations*. ORNL/TM-2005/39, Version 5, Vols. I-III, April 2005, Available from Radiation Safety Information Computational Center at Oak Ridge National Laboratory as CCC-725.

[13] Sell, C., "Global nuclear energy partnership." Foreign Press Center Briefing, 16 Feb. 2006. Washington, D.C.

[14] Stacey, W. M., *Nuclear Reactor Physics*. New York: John Wiley and Sons, 2001.